MVA2015 IAPR International Conference on Machine Vision Applications, May 18-22, 2015, Tokyo, JAPAN

Visual Detection and Species Classification of Orchid Flowers

Steven Puttemans & Toon Goedemé
KU Leuven, Campus De Nayer, EAVISE Research Group
Jan Pieter De Nayerlaan 5, 2860 Sint-Katelijne-Waver, Belgium
[steven.puttemans, toon.goedeme]@kuleuven.be

Abstract

The goal of this research is to investigate the possibility of using object categorization and object classification techniques in an industrial context with a very limited set of training data. As an industrial application of the proposed techniques we investigate the case of orchid flower detection and orchid species classification in an orchid packaging plant. Due to their large variety of colors and patterns, these orchid flowers are very hard to detect with classic segmentation based techniques but form an ideal test case for object categorization techniques. Due to the limited amount of training data available, we aim at building a system with close to no false positive detections but guaranteeing that each orchid plant still returns a single flower detection. Subsequently the detected flowers are passed towards a classification system of linear binary SVM classifiers trained on visual characteristics of the flowers. To increase the classification success rate, we combined results of single flowers, using majority voting, to reach an orchid plant based classification. The complete pipeline is optimized by effectively using the industrial application specific knowledge of the setup. By implementing this approach we prove that industrial object categorization and classification with high accuracy is possible, even if only a small training dataset is available.

1 Introduction

Robust object detection using object categorization techniques is a very active research topic. Previous research proved that these techniques are mostly demonstrated on very specific applications, like pedestrian or car detection, which are unrepresentative for industrial cases. Those state-of-the-art techniques mainly reach high accuracy by using enormous amounts of training data combined with efficient training algorithms. Our previous research [9] indicates that using those enormous datasets are not always needed when targeting industrial applications with certain scene or application specific constraints. We even suggested that a small training set can lead to a well performing application specific object detector. We want to prove this theorem by focusing on the industrial application of robust orchid flower detection and classification by using simple, easy to compute visual characteristics of the orchid flowers and exploiting the specifics of the setup. The choice of these simple visual features is mainly to reduce the computational effort.

This application has several challenges. With more than 100.000 different Phalaenopsis orchid flower cultivars, we notice a large intra-class variation in shape, size, color and pattern. Object categorization techniques can build a single model for flower detection based on this heavy variating object data. With only a limited set of training data available, we investigate the possibility of reaching high accuracy on the detection output. On the other hand the application has several setup-specific elements that make the training of a detector and classifier easier. First of all there is a controlled lighting, which restricts the intra-class appearance variation, and thus the amount of negative examples, and allows to use a color description for each species. Since color descriptions will not be unique for all those flower species, we will combine the color description with a appearance based classification of the flower, based on specific visual characteristics like texture. Secondly the position of the camera is known which leads to an effective reduction of the huge object search space based on the image pyramid [9]. For classification, we propose to look at a specific set of visual features like color differences, dotted patterns, radial lines, ... We efficiently combine all this information into a cascade of binary support vector machine classifiers that succeed in separating all provided orchid flowers into five texture based classes as seen in Figure 1. Combined with a unique color description this class label successfully separates all cultivars.

The remainder of this paper is organized as follows. Section 2 discusses related research while section 3 details how we detect orchid flowers by using object categorization. Section 4 elaborates on our flower species classification approach using binary support vector machines. Finally section 5 will wrap-up some conclusions and suggests future improvements.

2 Related Work

Many industrial object detection applications heavily rely on uniform lighting conditions and limited variation of the object class that needs to be detected, resulting in a threshold-based segmentation of the input data. In order to achieve this they place very hard constraints on the application setup. However in our case, the variety in flower species is huge (shape, pattern and color variation), which makes it impossible to use segmentation based approaches like [8]. As we suggested in [9], object categorization techniques, like [11, 2, 4, 3], can be used to tackle this problem. They have been used for robust object detection in situations with a lot of scene variation (lighting, occlusion, clutter, ...). In our case the situation is somewhat different, due

Figure 1: Example images of all five orchid flower classes. (a) Uniform (b) Lip (c) Striped (d) Spotted (e) Speckled

to the huge object variation (there are 60 Phalaenopsis species with in total more than 100.000 different cultivars) but with a uniform, controlled lighting and a known camera position. These application specific parameters can hopefully limit the amount of training data needed to obtain a robust object detector.

The work of Mathias et al. [6] clearly shows that the combination of object categorization and feature classification works for robust traffic sign classification. However in this case the lighting conditions are quite variable, but the color and shape features of the object itself are quite stable. This is just the opposite of our case, where the lighting is quite constant but the variation of the shape and color varies immensely.

The work of Nilsback and Zisserman shows related research on the multi-class classification of flower species [7] and the application of smart flower segmentation based on shape and color [8] although they do not have a controlled lighting and do not discriminate different cultivars of the same flower species.

More recently [10] describes an approach for efficiently classifying scanned leafs and orchid flowers, using an approach based on vantage feature frames. Their research is focused on a different variety of orchid flowers and due to the scanned image, the shape of the flower is far more unique than in our application. The work also suggests to create far more boundary conditions in order to ensure a successful classification and high recognition rate, which is exactly what we try to avoid, since they cannot be achieved in our case of industrial flower classification.

3 Orchid Flower Detection

A first step in processing orchid flowers is detecting where the actual flowers are in the input image, before they can be passed on towards the classification process, which will be discussed in section 4. Older but robust object categorization techniques like Viola and Jones [11] show good results on face detection whereas more recent techniques like deformable part models [4] and integral channel features [3] are mostly demonstrated on pedestrian detection. However none of these frameworks have been exhaustively tested for industrial object detection.

Since we have a limited set of training samples, and moreover a very limited set of positive training samples compared to a large set of negative samples, it was important to select an object categorization technique fit for this specific task. Using the HOG+SVM approach [2] showed not to work due to the very limited positive training sample set. The detector output gave indecent results and there was not enough data to generalize a good model. Also, the HOG+SVM approach largely depends on a balanced set of positive and negative training samples to increase accuracy, which was not the case for our application. The cascade classifier approach suggested by Viola and Jones [11] showed promising results on a very limited dataset, being able to generalize a model, even if a limited positive training set was available. In addition to that the framework uses features, being Haar wavelets or local binary patterns [5], that generalize better over a small set of data, whereas the HOG features do not generalize that fast. Therefore we took the basics of the framework and adapted it to our needs of generic object detection, and more specifically in this case the detection of orchid flowers. We combined the open source training and detection interface provided by OpenCV [1] and made a complete training and detection framework for any given object class.

Subsection 3.1 discusses the training of the object detector, where subsection 3.2 will elaborate on the actual detection task. Finally subsection 3.3 discusses the detection results.

3.1 Training the object model

In order to build a robust orchid flower model, we collected a set of training images. As positive object images, 252 orchid plant images were grabbed from the industrial pipeline which already has a fixed camera setup inspecting the plants. From these images, each flower was manually annotated with a bounding box, extracted from the original image and resized to an average size of 48×56 pixels. The original images, with the annotated pixels blacked out, were used as background images. From those images a set of controlled samples were grabbed as negative training examples. Due to the known illumination and camera position, an application specific detection model for our industrial case is created, which will not work in any other setups used for orchid detection. It is highly dependent on the used background setup and the way the orchids are presented to the system.

On each training image, the LBP or Haar wavelet features are calculated, for which weak classifiers are learned. The AdaBoost algorithm then decides which classifiers are the most discriminative on the training data. These classifiers are then added to the cascade of weak classifiers, until a desired level of detection accuracy was reached. The model is built by using 250 positive image samples of several orchid flower classes for each stage of weak classifier and 2000 randomly sampled negative image samples. The final model contains a set of 57 weak classifiers, each trained as a binary decision tree with a single depth layer. Weak classifiers are combined in stages until each stage reached a maximum false alarm rate of 0.5, while guaranteeing a minimum hit rate of 0.95 on the positive training data before moving on to the next stage. The training took 20 hours on a dual core processor with 8GB RAM of processing power. The combination of these parameters resulted in a final orchid flower model used for the detection part of the approach.

3.2 Object detection using the trained model

The general approach when using a cascade of weak classifier is to run through the whole image pyramid, a step-by-step downscaled version of the original input image, with a sliding window equaling the object size. This allows us to perform multi scale object detection using only a single scale object model. Unfortunately, the search space for object candidates turns out to be enormous when the size of input images grows. Based on the orchid flower model that was trained in the previous subsection and the knowledge of the camera setup in this industrial application, we retrieved a set of properties which can be used to effectively reduce the search space. In addition to reducing the search space

Figure 2: Orchid flower detections without false positives.

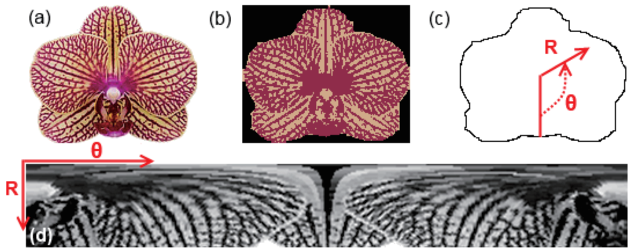

Figure 3: Different preprocessing steps needed for the feature calculation of each input window. (a) the segmented flower (b) K-means clustering (c/d) radial unwarping

by increasing the image pyramid scale step, we add restrictions to the object candidate size by limiting the scale range due to a fixed camera position. Reducing the search space of object candidates also helps to drastically reduce the amount of false positive detections. A constant and diffuse lighting leads to an effective foreground-background segmentation, again reducing the search space of object candidates. This results in a fragmented image pyramid, as discussed in [9].

A last improvement we made is related to the application specific requirements and can be achieved by applying a higher threshold on the last stage of the cascade of weak classifiers. Since we only aim to correctly classify a complete orchid, it doesn't matter if we miss some flowers in a single view. By combining multiple views of a single orchid for processing, we are able to increase the final stage threshold, so that only flower detections with a high certainty get accepted and no false positive detections occur anymore.

3.3 Detection results

The fact that we achieved a well performing object detector with a very limited training set, by carefully selecting the training parameters, is quite impressive.

On the complete validation set of 360 test images, not a single false positive detection was reported, while still maintaining at least a single detection on each image. An example of these false positive free detection outputs can be seen in Figure 2.

4 Orchid Type Classification

The previous section discusses the problem of accurately localizing the orchid flowers in the input image, however this does not yet solve the problem of identifying the exact Phalaenopsis flower cultivars. This section will present a complete pipeline for classifying flower cultivars towards a predefined class using a binary support vector machine setup and majority voting. Since there are so many cultivars, it is impossible to classify each single one of them. Based on the visual characteristics of the flowers, we divided the cultivars into five pattern based orchid classes. Combined with a color description of the flower itself we can then decide which cultivar is presented.

4.1 Visual characteristics and features

A first step in the flower classification pipeline is the segmentation of the flower from the background, based on the images retrieved from the orchid flower detector discussed in section 3. In our application we have the advantage of a known setup, including a blue background, which makes the segmentation easier. We also filter out any green regions which could be parts of the plants, like stokes or flower buds, partially occluding the actual flower. This segmentation leads to a segmented flower region image, which can then be passed on to the classification process. Histogram equalization is applied to the RGB channels to improve contrast.

Considering the five orchid flower classes mentioned in Figure 1, we made a set of observations about the flower's appearance. One general characteristic for all orchid flower classes is that if a flower has multiple color ranges, then there are at most two colors, called a foreground and a background color.

- **In the uniform color class**, the color of the background and foreground are almost equal.
- **The colored lip class** has a large color difference between background and foreground combined with a small amount of foreground blobs. Positioning of the foreground blob is at the bottom center of the flower.
- **The striped class** has a large color difference between background and foreground combined with a large amount of foreground blobs. The flower has strong radial edges.
- **The spotted class** has a large color difference between background and foreground combined with a small amount of foreground blobs. The foreground blobs have a random location.
- **The speckled class** has a large color difference between background and foreground combined with a large amount of foreground blobs. The edges are not dominantly radial.

Based on the visual characteristics of these 5 pattern based classes, we deduced a set of measurable features which can be used to train a decision classifier. Figure 3 visualizes all stages needed for gathering the necessary features from a single input image. We start by converting the image to the La*b* color space.

This is followed by several processing steps which lead to the specific flower features used for classification.

1. A K-means clustering (K=2) is applied on all pixels in the a*b* channels in order to define the foreground and background. This results into two clusters, each assigned the average cluster color. The cluster with the most pixels is the background. The first feature is achieved by calculating the color difference between foreground and background (Fig. 3(b)).

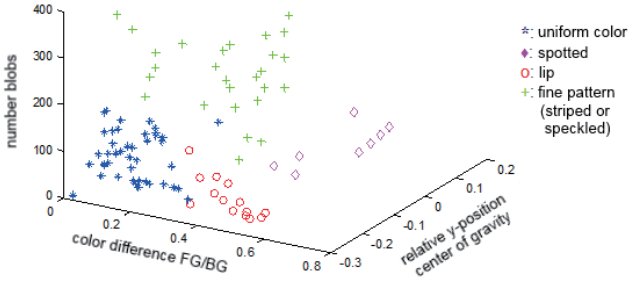

Figure 4: The feature space visualized with training samples of all classes.

2. The second feature is calculated as the relative y-position of the gravity center of the foreground.
3. A connected component analysis is applied on the binary clustered image. The ratio between the amount of blobs in foreground and background is stored as the third feature.
4. The radial dominant edges are quantified by applying a radial unwarping of the input image around its center (Fig. 3(c) and 3(d)). The ratio of the vertical and horizontal response to a corresponding Sobel filter is stored as the fourth feature.

4.2 Binary support vector machine tree

After calculating all the features of the training images, we built a set of binary SVM classifiers with linear kernels based on the training data. The use of other kernel types was impossible due to the limited training data set available which would result in very weak performing classifiers.

Figure 4 shows the distribution of the feature space for the different flower classes that we want to separate by training a set of linear SVM classifiers. Based on this distribution, a binary decision tree based on four linear SVM classifiers was built as seen in Figure 5. This structure uses a set of intermediate classes to store in between results, like the pattern class which is subdivided into a coarse and a fine pattern class.

The complete binary tree of linear SVM classifiers is trained on a set of 97 training images of different orchid flower species, of which all four features are calculated and used for training the different binary SVM classifiers.

4.3 Classification results

After producing the complete classification pipeline, a validation set of orchid flowers was gathered containing 115 flower images of different species. The classification results on this validation set can be seen in Table 1. Keep in mind that the classification results are based on a single flower image level.

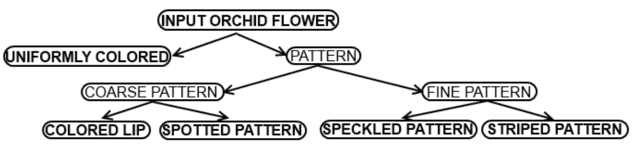

Figure 5: Scheme of the binary SVM classifier tree.

Table 1: Classification result on the limited validation set.

Class	Amount	Correct
Uniformly Colored	51	94.23%
Colored Lip	16	93.75%
Spotted Pattern	10	100%
Speckled Pattern	16	100%
Striped Pattern	23	78.26%

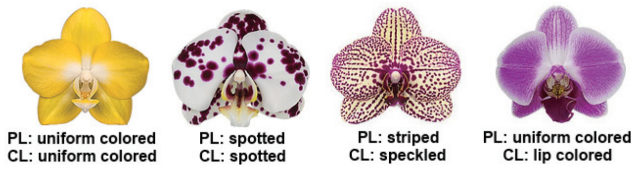

Figure 6: Professional (PL) versus classifier (CL) labels.

As a last step we can combine the single flower classification results together with the detection pipeline to increase the classification certainty of a single orchid plant scan, since we only require a single label for each plant. For this we use majority voting based on all single flower classifications in order to reach a robust classification result for the orchid itself. This helps to ensure that if a single flower raises a wrong classification, the overall label of the orchid plant is still correct.

Overall the results are very good regarding the limited data used and the great variability of the object. However, we noticed that due to the large intra-class variance in appearance, obtaining a 100% correct classification result for each flower image will be near to impossible, especially when increasing the test set. Even domain specific experts have problems of dividing all flowers into the correct ground truth classes (see Figure 6).

4.4 Detection and classification combined

Next to the neat testing images used in 4.3, which were very constrained (white background, orchids in uniform position, single flowers, ...), we also performed tests on real images retrieved from our industrial setup using the orchid flower detector. Our application of orchid flower grading has a very unique setting with a constant blue background and nothing else but flowers and stokes of the plant in the captured images. By applying a small color based filter for both blue (background) and green (stokes and flower buds) color ranges in each detection window, we successfully remove the background clutter and replace it by a white background like the test samples used in section 4.3. This ensures that this background clutter doesn't influence the feature calculation process. Figure 7 shows several flowers that were cut out from a single orchid plant and then segmented using the aforementioned approach and which were then successfully classified.

We acknowledge that there are two specific cases where our simple flower-background segmentation approach yields no clean result. The first case is where the background of a detected flower is a combination of multiple other flowers. However since all the flower of a single orchid plant have similar colors, this doesn't influence the feature calculation drastically. Secondly when flowers are too tilted and the viewpoint

Table 2: Classification result on complete orchid plants.

Orchid	Manual	#Flowers	Uniform	Lip	Spottted	Speckled	Striped	Majority
1	Striped	10	1	0	0	0	9	Striped
2	Uniform	11	11	0	0	0	0	Uniform
3	Speckled	16	1	0	5	3	7	Speckled

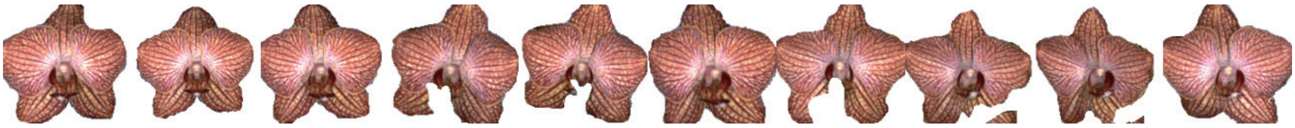

Figure 7: Example of orchid flower detections being filtered for type classification - Orchid 1 from Table 2.

changes too much, then the classification process can yield wrong results. We try to avoid these cases by only training the orchid flower detection model with flower samples that have the desired viewpoint and by increasing the detection threshold so that it is high enough to simply ensure that the correct viewpoint is returned in most detections. In all cases where the classification goes wrong, majority voting of single flower classification results solves these issues and guarantee that a total orchid plant classification is still correct. Table 2 illustrates this process of majority voting by supplying 3 orchid plants to our complete pipeline. First each plant yields several orchid flowers, by performing a multi-scale detection using our pretrained orchid model, which are then segmented from the background as suggested and passed to the classification pipeline. By using majority voting each plant is labeled the correct class.

5 Conclusion

We investigated the possibility of accurately detecting orchid flowers and classify them into five larger visual texture classes. For the detection we suggested using a Viola and Jones based approach with LBP features and AdaBoost learning, where the classification part was solved by training a binary tree of linear SVM classifiers on a limited set of training data. We conclude that we successfully built a robust orchid flower detection and classification pipeline that reaches a desired accuracy. By smartly combining the classification output using a majority voting system we achieve a very high classification success rate on the level of a single orchid plant. Knowing that we had a very limited training data set for both the detection model training and the training of the SVM classifiers, the end result is quite impressive compared to the current research in object detection, where still multiple thousands of training samples are used to reach robust classifiers. An explanation for this can be found in the controlled lighting conditions and the known setup (camera position, possible background, orchid flower position).

In the future we plan to further investigate this problem, and mainly the influence of adding more training data to both the training of the object detection model and the SVM classifier structure. This will also allow to use more complex non-linear kernels in the SVM training, which allows to better separate the classes from each other and reach a higher accuracy.

We acknowledge that the created system is easy adaptable for any industrial detection problem and that the benefit of using a limited set of training data, leading to a decent accuracy, allows us to extend this approach. We will apply the same technique to detecting flower buds to ensure that similar results can be achieved.

6 Acknowledgements

This work is supported by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT) via the IWT-TETRA project *TOBCAT: Industrial Applications of Object Categorization Techniques*. We would also like to thank Aris BV, the company that provided the orchid flower datasets.

References

[1] G. Bradski and A. Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc., 2008.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886–893. IEEE, 2005.

[3] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, volume 2, page 5, 2009.

[4] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, pages 1–8. IEEE, 2008.

[5] M. Heikkila, M. Pietikainen, and C. Schmid. Description of interest regions with local binary patterns. *Pattern recognition*, 42(3):425–436, 2009.

[6] M. Mathias, R. Timofte, R. Benenson, and L. Van Gool. Traffic sign recognitionhow far are we from the solution? In *IJCNN*, pages 1–8. IEEE, 2013.

[7] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008.

[8] M.-E. Nilsback and A. Zisserman. Delving deeper into the whorl of flower segmentation. *Image and Vision Computing*, 2009.

[9] S. Puttemans and T. Goedemé. How to exploit scene constraints to improve object categorization algorithms for industrial applications? In *VISAPP*, volume 1, pages 827–830, 2013.

[10] A. R. Sfar, N. Boujemaa, and D. Geman. Vantage feature frames for fine-grained categorization. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 835–842. IEEE, 2013.

[11] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, pages I–511, 2001.